Practical assignment 4 (R04) (Building your first deep neural network on page 130 of the Grokking Deep Learning textbook)

Consider the following code snippet which implements the backpropagation learning algorithm:

```
import numpy as np
np.random.seed(1)

def relu(x):
    return (x > 0) * x

def relu2deriv(output):
    return output>0

streetlights = np.array( [[ 1, 0, 1 ], [ 0, 1, 1 ], [ 0, 0, 1 ], [ 1, 1, 1 ] ] )
walk_vs_stop = np.array([[ 1, 1, 0, 0]]).T

alpha = 0.2
hidden_size = 4

weights_0_1 = 2*np.random.random((3,hidden_size)) - 1
weights_1_2 = 2*np.random.random((hidden_size,1)) - 1

for iteration in range(60):

    layer_2_error = 0

    for i in range(len(streetlights)):

        layer_0 = streetlights[i:i+1]
        layer_1 = relu(np.dot(layer_0,weights_0_1))
        layer_2 = np.dot(layer_1,weights_1_2)
        layer_2_error += np.sum((layer_2 - walk_vs_stop[i:i+1]) ** 2)
        layer_2_delta = (layer_2 - walk_vs_stop[i:i+1])
        layer_1_delta=layer_2_delta.dot(weights_1_2.T)*relu2deriv(layer_1)

        weights_1_2 -= alpha * layer_1.T.dot(layer_2_delta)
        weights_0_1 -= alpha * layer_0.T.dot(layer_1_delta)

    if(iteration % 10 == 9):
        print("Error:" + str(layer_2_error))
```

Please modify the code mentioned above in the following manner:

1) Add a second hidden layer with 6 nodes.
2) Train the neural network for 1024 iterations.

Please write a report of no more than 1000 words on neural network training. Cite reputable sources to motivate your arguments. Submit your Python (.py) text file and Word document as two separate assignments on eFundi.